IEEE Open Journal of
Intelligent Transportation
Systems

Received 23 September 2020; revised 1 December 2020; accepted 28 January 2021. Date of publication 3 February 2021;
date of current version 12 April 2021.

Digital Object Identifier 10.1109/OJITS.2021.3056925

A Platform-Based Incentive Mechanism for Autonomous Vehicle Crowdsensing

ALIREZA CHAKERI[1] (Member, IEEE), XIN WANG[2] (Student Member, IEEE), QUENTIN GOSS[3] (Graduate Student Member, IEEE), M. ILHAN AKBAS[3] (Member, IEEE), AND LUIS G. JAIMES[2] (Member, IEEE)

[1]Department of Computer Science, University of South Florida, Tampa, FL 33620, USA

[2]Department of Computer Science, Florida Polytechnic University, Lakeland, FL 33805, USA

[3]Department of Electrical, Computer, Software and Systems Engineering, Embry–Riddle Aeronautical University, Daytona Beach, FL 32114, USA

CORRESPONDING AUTHOR: L. G. JAIMES (e-mail: ljaimes@floridapoly.edu)

This work was supported by the National Science Foundation (NSF) under Award 1739409.

ABSTRACT In this article, we present an incentive mechanism for Vehicular Crowdsensing in the context of autonomous vehicles (AVs). In particular, we propose a solution to the problem of sensing coverage of regions located out of the AVs' planned trajectories. We tackle this problem by dynamically modifying the AVs' trajectories and collecting sensing samples from regions otherwise unreachable by originally planned routes. We model this problem as a non-cooperative game in which a set of AVs equipped with sensors are the players and their trajectories are the strategies. Thus, our solution corresponds to a model in which expected individual utility drives the mobility decision of participants. Using open-street maps, SUMO vehicular traffic simulator, and extensive simulations, we show our algorithm significantly outperforms traditional approaches for trajectory generation. In particular, our performance evaluation shows a significant lift in crowdsourcer coverage, road utilization, and average participant utility.

INDEX TERMS Vehicular crowd-sensing, intelligent mobility, cyber-physical systems, game theory.

I. INTRODUCTION

THE RAPID spreading of mobile computing has opened the door to a new data collection paradigm called mobile crowdsensing (MCS) [1], [2]. MCS takes advantage of the massive use of smart phones and users' daily mobility patterns to collect sensing samples from places otherwise unreachable using traditional static sensing methods. Typical MCS applications include the collection and report of environment variables such as pollution [3], temperature, pollen, noise [4] and so on. In all these cases, the idea is to collect a representative set of sensing samples in order to re-construct a variable of interest, namely spatial coverage. In addition, sampling from a variable with high levels of variability such as temperature and pollution in which factors such as winds might change its value often; requires sampling at regular time intervals, namely temporal coverage.

The review of this article was arranged by Associate Editor Sabah Mohammed.

Similar to MCS, Vehicular Crowdsensing (VCS) is a new data collection paradigm in which the sensing task is based on sensing trajectories rather than on isolated discrete sensing samples [5]. A typical VCS system consists of two roles: a set of participants and one or several crowdsourcers [6]. Here, participants are a group of vehicles willing to contribute with a set of sensing tasks, and the crowdsourcers are the ones posting those sensing tasks. Examples of VCS tasks include route and infrastructure monitoring [7], traffic prediction [8], and map creation and updating [9]. VCS leverages vehicles' mobility patterns which usually cover wider regions than pedestrians', making possible to acquire sensing samples from places otherwise unreachable. On the another hand, unlike MCS, vehicles' mobility patterns are usually predictable [10], which means that any sensing task has to be located in the already known participants' trajectories. This constraint limits the acquisition of sensing samples to highways, and popular roads, clearly a non representative

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

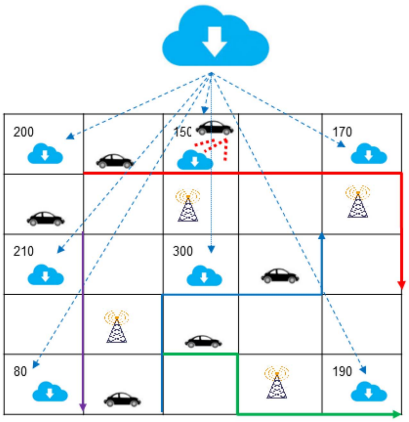


FIGURE 1. Grid Model.

set of samples. Thus, the use of these samples may result in a poor reconstruction of the variable of interest.

We formulate this problem in the form of the following question: Is it possible to use VCS for acquisition of sensing samples from regions located out of the participants' pre-planned trajectories? We tackle this challenge by designing an incentive mechanism that dynamically detours vehicles from their initial trajectories. We call our mechanism the Autonomous Vehicular Crowdsensing Game (AVCG). We model the interaction between AVs as a non-cooperative game in which the sensing trajectories are the AVs' strategies. Further, AVCG finds the approximated trajectory equilibrium for AVs. This equilibrium corresponds to the set of stable sensing trajectories resulting from the best response of each AV to other AVs' local sensing trajectories. We call this approximated solution as Approximated Trajectory Nash Equilibtium (ATNE). Advantages of ATNE include the maximization of AVs' utilities given other AVs' behaviors and improvements on road network.

Figure 1 sketches a simplified version (without a street network overlay) of our proposed AVCG system. Here, the region of interest is represented by a square grid. This region might correspond to an urban region in which a set of AVs equipped with sensors have the ability of sensing and reporting while performing their daily commute. Here, some grid cells represent crowdsourcers which are in charge of posting a sensing task in their region of influence, and the associated task reward. These announcements attract a crowd of vehicles who are willing to work on the posted tasks, namely visiting the requested regions (grid cells) and collecting sensing samples from those cells. Thereby, the interested participants submit to the platform their current locations, destination, and the amount of data they will collect (sensing plan). In this work the terms user, participant, player, contributor, AV, and vehicle are used indistinctly. The same happens with terms vehicular crowdsensing (VCS), and autonomous vehicular crowdsensing (AVCS).

The followings items summarize this article's main contributions:

- We present a behavioral model for AVs' vehicular crowd-sensing.
- We formulate the VCS trajectory Nash Equilibrium (NE) problem.
- We design a greedy algorithm to approximate VCS trajectory NE.
- We evaluate our algorithms using a real-world traffic maps, and the state of the art traffic simulator SUMO

The rest of this article is organized as follows. Section II corresponds to the related work. We present the problem formulation, and our greedy approach to solve it in Sections III and IV respectively. Section V presents this article's performance evaluation, and finally, Section VI concludes this article.

II. BACKGROUND AND RELATED WORK

This section presents a brief background of some of the topics addressed in this work. These include Mobile Crowdsensng (MCS), Vehicular Crowdsensing (VCS), and application of game theory to crowdsensing. The section also presents the most related approaches to the proposed method and its differentiating properties. Spatial or Mobile Crowdsensing (MCS) is usually identified with systems in which pedestrians (participants) from diverse communities use their smartphones for data collection, and reporting [2]. The mobility component relies on the mobility patterns of the stakeholders [11]. Authors such as Tong *et al.* [12] point out the unique characteristics of MCS, its challenges, and core issues such as task assignment, quality control, incentive mechanism, and privacy protection.

Task assignment allocation has also been studied from centralized and distributed perspectives. Particular attention has received the case of transportation networks. Here, the centralized platform tells each driver which path to take in order to minimize the average travel time, while in the distributed version, each driver chooses its own path. Unfortunately, the selfish behaviors of drivers in a distributed environment may produce counter-productive effects. One of the first studies showing the situation in which assigning more resources (roads on a network) may actually decrease the efficiency of the network is Braess *et al.* [13] in the Braess Paradox.

In a recent study, Li and Morse [14] presents the Power Allocation Game (PAG) which is better known as the network paradox. Unlike Braess paradox, PAG takes place in the context of geopolitics. Here, countries are modeled as nodes of a network in which there are friends and adversaries. Thus, a country allocates resources to support friends and demise adversaries in the hope of maximizing the country welfare. However, the authors show that increasing the number of allies may actually decrease a country's total welfare in equilibrium, especially when these allies have conflicts among themselves. A common way to compare the efficiency between a centralized versus a distributed system is "The Price of Anarchy" [15], which in the case of transportation networks, corresponds to the ratio of average travel time computed by using a centralized and a distributed approach.

Although most of the crowdsensing research focuses on the interaction among selfish players in competitive environment, some authors such as Xue *et al.* [16], and Jaimes *et al.* [17] focus on the conditions in which mutual cooperation maximize the long-term reward. In case of Xue *et al.* [16] the authors found that a strategy based on cooperation maximizes the long-term reward when participants play the iterated prisoner's dilemma in scale-free networks. In the case of Jaimes *et al.* [17] the authors propose a set of negotiation algorithms in the context of recurrent reverse auctions. Here, the winners of an auction round propose deals to their loser neighbors, and losers accept the deal from one or more winners if, and only if the sum of the offers is greater than its return of investment. Here again, the authors show that a cooperative behavior results in a higher reward.

Game theory has been widely used as a framework to study the interaction among selfish participants in crowdsensing networks, and in particular, the concept of equilibrium among which the most common is the Nash equilibrium. For instance, Lu *et al.* [18] use a non-cooperative game to analyse the inefficiency of two-sided rating protocol for crowdsourcing service exchange. Here, participants play the dual role of service requester, and server (service providers). Since providing services incurs costs, rational selfish participants are inclined to provide low level services and they avoid the role of server. This unbalanced role structure (i.e., requester-server) caused by rational selfish non-cooperative users, results in a decrease of the social welfare. Non-cooperative games have been also used to model the problem of channel access control for cooperative vehicle safety systems [18]. The problem originates from the trade-off between vehicles broadcasting massive information to communicate others, and the need of real time access to the channel for tracking other vehicles and avoid collisions.

Non-cooperative games have been also used to model incentive mechanisms for MCS. For instance, Yang *et al.* [19] introduce the sensing game. This incentive mechanism takes place in scenarios in which a crowdsourcer posts a sensing task and its corresponding compensation, and set of interested participants submit their corresponding sensing plans, namely the units of time they are willing to work on that task. They present the game's solution as the Nash Equilibrium (NE) sensing plan and provide an algorithm for its computation as well as a proof of the solution's uniqueness. Chakeri and Jaimes [20], [21] extend the work of Yang to multiple crowdsourcers, and games with incomplete information under some conditions. Although, these works provide an intuition about the possible solution of the AVs' trajectory Nash equilibrium (presented in this article), their combinatorial nature and complexity make them unfeasible to reach an analytical solution. Unlike our ATNE approach, Yang's and Chakeri's work corresponds to a one-shot solution for a crowdsourcer, while our approach builds trajectories over time using several crowdsourcers and several participants. We use Yang's result as part of our local level approach (Sensing Plan Adjusting (SPA) algorithm), and a greedy strategy to build the participant trajectories. The interested reader can find a comprehensive review of several game theoretical methods applied to crowdsensing here [22], [23].

In addition, the emergence of Internet of Vehicles in the big data era is opening new possibilities for extending traditional MCS based on pedestrians to terrestrial and aerial vehicles. This augmented data collection paradigm is called vehicular crowdsensing (VCS). Several studies [24] have predicted that by the end of 2020 one fifth of the vehicles on road will have Internet connection, more than 200 sensors, and will generate around 4000 GB data every day. All of the data traffic supported by high speed vehicular networks (VANETS) [25] will make possible the efficiency and safety of the road network. A pioneer paper on VCS corresponds to the work of He *et al.* [5], which presents a framework for participants recruitment. Both a greedy and a genetic algorithm are proposed to provide sensing coverage based on a subset of the predicted participants' sensing trajectories. This work has inspired other works such as [6], [26] which present a different solution of the same optimization problem (coverage by using participant sensing trajectories). The techniques in the aforementioned VCS were extended by several approaches [27], [28] with the idea of providing sensing coverage based on a combination of participants' trajectory segments rather than entire trajectories. In these cases, the platform only acquires the AVs' trajectory segments that increase coverage, avoid data redundancy and improve budget utilization. However, unlike our proposed ATNE, none of these approaches provide sensing coverage for regions out of the participants' planned trajectories.

Following a different approach, the work of Zhu *et al.* [10] uses multi-robot planning concepts for generating a set of trajectories that improves data coverage. However, these trajectories are computed beforehand and suggested to participants. Although, their work is similar to our approach in the sense that it modifies participants original trajectories for improving coverage, but it provides no behavioral model for AVs and, therefore, no detailed explanation about the incentive mechanism for the acquisition of trajectory sensing data.

III. SYSTEM MODEL

In this section, we present the components of the AVCG, and show their interconnection. Elements of AVCG include a set of AVs traversing in a set of sub-regions for sensing tasks, and a set of crowdsourcers. Some sub-region of interests have associated crowdsourcers who post rewards in the hope of attracting workers (AVs). All AVs working for a crowdsoucer's sensing task compete with each other to maximize their own utilities. In this scenario, AVs are willing to figure out their most profitable trajectories going from their starting locations to their destinations.

We assume that all vehicles are autonomous, namely passenger decision making is not involved. Also, participants

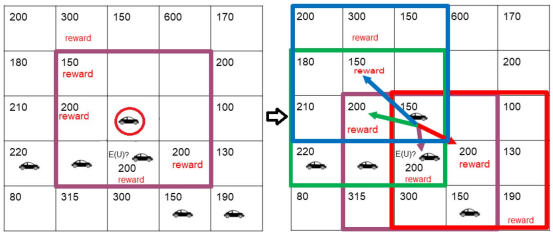


FIGURE 2. Building a trajectory.

provide departure and destination locations, and the shortest path between them are the participants' default trajectories. We also assume AVs are allowed to freely plan their routes and deviate if they are sufficiently incentivized. The system is assumed to be synchronous and all participants move one step at a time. We used a discretized version of the area in the form of uniformly shaped square cells where their resolution can be adjusted according to the granularity of the sensing. The AVs can only visit cells which are connected by roads that cross their geographical boundaries.

Before describing in detail the problem formulation, we present a motivating example to visualize the problem. Figure 2 (left) shows a grid representation of a region of sensing interest. Here, there are some posted sensing tasks and their corresponding rewards. In this simplified scenario (without an overlaid map), a set of vehicles are traversing a region of interest and greedily collect as much reward as they can.

Consider the participant located in cell (3, 3) who wants to select its next move. A naive participant may select any of the cells with a reward of 200. While these cells have the highest task rewards, there is also a competition among the vehicles in those cells' adjacent neighborhoods (Figure 2 (right) red, purple, and green boxes). On the another hand, the participant has the option to traverse to the north west cell with a 150 reward as there is minimal competition there (blue box) that allows it to get a maximum utility. This is because the participant will be the only AV able to reach there at the next time step.

A. PROBLEM FORMULATION

Our designed VCS problem consists of multiple crowdsourcers and participants. Crowdsourcers are whether distributed independently across the set of sub-regions $G = \{g_1, g_2, \ldots, g_T\}$ or work for a central platform. The crowdsourcers post rewards to attract various participants to collect data. Let $C = \{c_1, c_2, \ldots, c_F\}$ be the set of F crowdsourcers located at different F sub-regions. Note, if a sub-region does not have a crowdsourcer, the associated reward is set to zero. Also, let $V = \{v_1, v_2, \ldots, v_M\}$ be the set of M participants, where $S = \{s_1, s_1, \ldots, s_M\}$ and $D = \{d_1, d_2, \ldots, d_M\}$ represent the sets of their starting and destination sub-regions. Each crowdsourcer c_j announces a sensing task in its area of influence, and the corresponding task reward R_j.

We consider AVs as a set of rational players who are attracted by the posted rewards if they get a positive utility. The utility of the ith participant who works for crowdsourcer j is shown in Equation (1).

$$u_i^j = \frac{t_i^j}{\sum_{i \in W_j} t_i^j} R_j - k_i^j t_i^j \tag{1}$$

where W_j is the set of participants working on the sensing task posted by the crowdsourcer j with cardinality of w_j, k_i^j is the cost in which i incurs for working for crowdsourcer j and t_i^j is the sensing plan, namely the amount of data that participant i will be collecting for crowdsourcer j. The goal of the contributors is to maximize their corresponding utilities by submitting the optimal sensing plans. On the other hand, the contributors' sensing plans provide a utility for the crowdsourcers that can be generally written as

$$\bar{u}_j = g_j\left(t_1^j, t_2^j, \ldots, t_M^j\right) - R_j, \tag{2}$$

where $g_j(.)$ is the jth crowdsourcer valuation function of contributors' sensing plan. We assume that $g_j(.)$ is a strictly concave function in variables $t_1^j, t_2^j, \ldots, t_M^j$. This general assumption has been adopted in the literature. In particular, the authors in [19] set $g_j(.)$ to $\lambda_j log(1 + \sum_{i \in W_j} log(1 + t_i^j))$, where λ_j is a system parameter. The inner log term reflects the crowdsourcer's diminishing return on the work of the contributor i, and the outer log term reflects the crowdsourcer's diminishing return on participating contributors.

When the crowdsourcer posts a reward, a game takes place between the contributors in that sub-region. This game is called the *sensing game*. In the sensing game, the involved participants provide sensing plans. An example of sensing plan could be the number of sensing samples that a participant provides to estimate the levels of pollution. Now the natural question that rises is how to find the sensing plans for the contributors such that none of the contributors has anything to gain by unilaterally changing its sensing plan. In fact, this stable configuration corresponds to the concept of Nash equilibrium in game theory. A Nash equilibrium is a strategy profile such that each player's strategy is a best response to the other players strategies. Specifically, the authors in [29] proved that the sensing game has unique Nash equilibrium sensing plans, and provided a linear time algorithm to compute the only equilibrium. Since we make reference often to the proposed algorithm, it is rewritten here for the ease of the reader.

We use Algorithm 1 to compute the sensing plans for a set of participants, each incurring in a different cost k to work for a crowdsourcer. Therefore, participant's utility is depended of aforementioned sensing plan and computed by Equation (1). Consequently, the total utility of participant i for his journey from its starting location s_i to its destination d_i can be computed as $u_i = \sum_{g_k \in p_i} u_i^k$, where p_i is its entire trajectory. Thus, each participant builds its own optimal trajectory that maximizes its total utility as a result of all local competition that he participated. We model these interactions

Algorithm 1: Computation of the Nash Equilibrium Sensing Plans for Participants in Crowdsourcer j

1: Sort the set of contributors W_j ($|W_j| = w_j$) in crowdsourcer j according to their costs, $k_1^j \leq k_2^j \leq \cdots \leq k_{w_j}^j$;
2: $H \leftarrow \{1, 2\}, i \leftarrow 3$;
3: **while** $i \leq w_j$ and $k_i^j \leq \frac{\sum_{l \in H} k_l^j}{|H|}$ **do**
4: $\quad H \leftarrow H \cup \{i\}, i \leftarrow i + 1$;
5: **end while**
6: **for all** $i \in W_j$ **do**
7: $\quad$ **if** $i \in H$ **then**
8: $\quad\quad (t_i^j)^* = \frac{(|H|-1)R_j}{\sum_{l \in H} k_l^j}\left(1 - \frac{(|H|-1)k_i^j}{\sum_{l \in H} k_l^j}\right)$;
9: $\quad$ **else**
10: $\quad\quad (t_i^j)^* = 0$;
11: $\quad$ **end if**
12: **end for**
13: **return** $((t_1^j)^*, (t_2^j)^*, \ldots, (t_{w_j}^j)^*)$;

between the participants as a non-cooperative game, where the players are the AVs and their strategies are their possible trajectories. We call the modeled game as Autonomous Vehicular Crowndsensing Game (AVCG). Formally speaking, consider the game $<V, (P_i), (u_i)>$, where V is the set of AVs, P_i is the set of trajectories available to v_i, and $u_i : \prod_{j=1}^{M} P_j \rightarrow R$ is the set of possible payoffs for v_i. Also, $P = \prod_{j=1}^{M} P_j$ denotes all possible combination of trajectories. Each element of P_i is designated by p_i, and p_{-i} represents the set of players' strategies except for the player i. We call each $p = (p_1, p_2, \ldots, p_M)$ a configuration of the modeled game. Now, we are interested in finding the set of trajectories in which none of the players has anything to gain by unilaterally changing its current trajectory, i.e., the Nash equilibrium configuration. Particularly, $p^* = (p_1^*, p_2^*, \ldots, p_M^*)$ is a Nash equilibrium configuration if, $\forall i$, p_i^* is a best response to p_{-i}^*. We say p_i^* is a best response to p_{-i}^* if $u_i(p_i^*, p_{-i}^*) \geq u_i(\hat{p}_i, p_{-i}^*)$, $\forall \hat{p}_i \in P_i$.

Because of the computational complexity of finding the strategic Nash equilibrium configuration in AVCG and the exponential number of possible trajectories for each player, in this article, we introduce a tractable greedy solution.

B. EXPECTED UTILITY AND EXPECTED SENSING PLAN

In this greedy environment, contributors make decisions about finding their next positions by locally evaluating their possible next movements given all of the available information in the current time step. Thus, it is reasonable to assume a rational participant will select the move to the sub-region that yields the highest expected utility among all of its possible available sub-regions. Equation (3) represents the expected utility (EU) of moving to a target sub-region evaluated by AV i.

$$EU_i = \sum_{A \subset V, i \in A} \prod_{j \in A} P_j \times \prod_{j \notin A} (1 - P_j) \times u_i|A \tag{3}$$

Here, V is the set of AVs in the adjacent cells of a given crowdsourcer, and $A \subset V$ is the subset of those AVs visiting that crowdsourcer at the same time with $i \in A$. In addition, $P_j = \frac{1}{q_j}$ is the probability of AV j to reach the area of influence of the crowdsourcer, where q_j is the number of potential cells that AV j can visit at the next time step. As we mentioned in the system model, a cell can be visited if there is a road segment connecting the cell to the participant's location. Finally, $u_i|A$ is the utility i gets, when participants in A are the area of influence of that crowdsourcer.

Similar to Equation (3), Equation (4) shows the Expected Sensing Plan (ESP), i.e., the expected amount of data a participant will be collecting for a given crowdsourcer. However, here $t_i|A$ is the sensing plan of the ith participant computed when participants in A are in the region of influence of that crowdsourcer.

$$ESP_i = \sum_{A \subset V, i \in A} \prod_{j \in A} P_j \times \prod_{j \notin A} (1 - P_j) \times t_i|A. \tag{4}$$

C. SENSING CAPACITY

In order to prevent the unrealistic behavior of infinite movements and monetary collections, we add a collection constraint for each participant over his total journey. These constraints could be imposed by the participants ability to collect data based on their Internet provider data plan, the device remaining power and etc. In fact, the participants deviations from their original trajectory, i.e., the shortest path toward their destinations, are controlled by the sensing data capacity. However, the cost of deviation could also encompass other factors such as vehicle gas, participant collection time allowance and etc. Any of these factors will add another layer of constraints on the participants that will make the model more realistic. In this article, we only consider the sensing data capacity imposed on the participants. In particular, let's $B = \{b_1, b_2, \ldots, b_M\}$ represents the sensing capacities of the participants. Once a participant's sensing capacity is totally consumed, the participant stops collecting and heads to its final destination.

IV. GREEDY ALGORITHM FOR ATNE

In this section, we present our Approximated Trajectory Nash Equilibrium (ATNE) algorithm splitted into three components: next location selection, participant trajectory development, and sensing plan adjustment.

A. NEXT LOCATION SELECTION

Let's consider the ith participant (v_i) who evaluates moving to any of its z adjacent valid neighbors. A neighbor is valid if there is a road connecting that place with the participant's current location. The inputs of the Algorithm 2 include the v_i's capacity (b_i), the list of crowdsourcers in v_i's neighborhood, and number of AVs in v_i's neighbors' neighborhoods. The algorithm outputs the v_i's next location.

It loops through v_i's neighborhood (line 4) and computes the expected utilities (EU) and expected sensing plans (ESP) of them. Then, it moves to the location c with the highest EU as long as its sensing capacity (b_i) has enough room to accommodate the expecting sensing plan for c, namely

Algorithm 2: Selecting Next Place to Visit by Participant v_i

input : sensing capacity b_i, valid cells C= $\{1, 2, \ldots, z\}$, rewards $\{R_1, R_2, \ldots, R_z\}$, number of participants in the neighborhood of the valid cells $\{n_1, n_2, \ldots, n_z\}$, memory size q, cost of sensing in cell c is k_c
output: selected next location: c

1: $next \leftarrow none$;
2: $Max \leftarrow 0$;
3: $STM.enqueue(\emptyset)$
4: **for** $c \in C$ **do**
5: $Counter++$
6: **if** $R_c == 0$ **then**
7: *continue*;
8: **else if** $R_c > 0$ & $n_c == 1$ **then**
9: $ESP \leftarrow 1$;
10: $EU \leftarrow R_c - k_c$;
11: **else**
12: $EU = \sum_{A \subset V, i \in A} \prod_{j \in A} P_j \times \prod_{j \notin A} (1 - P_j) \times u_i | A$
13: $ESP = \sum_{A \subset V, i \in A} \prod_{j \in A} P_j \times \prod_{j \notin A} (1 - P_j) \times t_i | A$
14: **end if**
15: **if** $((EU \geq Max)$ & $(b_i \geq ESP)$ & $(c \notin MB))$ **then**
16: $Max \leftarrow EU$
17: $next \leftarrow c$
18: MB.enqueue(c)
19: **if** $(Counter \bmod q) == 0$ **then**
20: $STM.dequeue()$
21: **end if**
22: **end if**
23: **end for**
24: **if** $next == none$ **then**
25: $next \leftarrow SelectWeightedRandom$
26: **end if**
27: **return** *next*

Algorithm 3: Participant Trajectory Development

input : sensing capacity b_i, valid cells C= $\{1, 2, \ldots, z\}$
output: Participant v_i trajectory

1: $trajectory.push(s_i)$
2: $L \leftarrow trajectory.lastElement$
3: **while** $L \neq d_i$ **do**
4: $c \leftarrow SelectingNextPlaceToVisit$;
5: $sp \leftarrow AdjustingSensingPlan$;
6: **if** $sp \leq b_i$ **then**
7: $b_i = b_i - sp$
8: **end if**
9: $trajectory.push(c)$
10: $L \leftarrow trajectory.lastElement$
11: **end while**
12: **return** *trajectory*;

Algorithm 4: Sensing Plan Adjustment

input : v_i's sensing capacity b_i, and capacities of other AVs $\{b_1, b_2, \ldots, b_{(w_c-1)}\}$ arriving at the same cell c, and sensing costs k_i's, and cost of other AVs sensing cost $\{k_1, k_2, \ldots, k_{(w_c-1)}\}$
output: actual sensing plan *asp*

1: $asp \leftarrow 0$
2: **while** **do**
3: $sp \leftarrow Algorithm\ 1$
4: $counter \leftarrow w_c$;
5: **for** $j \in W_c$ **do**
6: **if** $sp_j \leq b_j$ **then**
7: $Counter++$;
8: **else**
9: $W_c \leftarrow W_c - \{j\}$
10: $w_c = |W_c|$
11: **end if**
12: **end for**
13: **if** $w_c == counter$ **then**
14: break
15: **end if**
16: **end while**
17: $asp \leftarrow sp$
18: **return** *asp*

if $b_i \geq ESP$. If v_i is the only participant in the neighborhood of c then ESP is set to 1 (line 8). By replacing this value of ESP in Equation (1), the utility for the only participant working for that crowdsourcer becomes $R_c - k_c$. On the other hand, if there are several participants in c's neighborhood, the expected sensing plan is computed using Equation (4) (line 13). To avoid data collection redundancy, the final *if* statement (line 15) includes a mechanism that prevents participants from immediately returning and collecting from places from which they were already collecting at q-previous time steps. We call this mechanisms a Memory Buffer (MB), and q is a system parameter. If there are not crowdsourcers in v_i's neighborhood or if v_i's capacity is not enough to accommodate the sensing plans computed for any of its neighbors or if the selected location is in participant's BTM, then the next location to visit is computed by using method *SelectWeightedRandom* (line 25). This function uses the shortest distances from each v_i's neighbor to v_i's destination to generate weights for the random selection of the next location. In other words, the function gives higher chance to neighbors located closer to v_i's destination to be selected.

B. PARTICIPANT V_I TRAJECTORY

Algorithm 3 uses Algorithm 2 to move v_i until it reaches its destination. In particular, once v_i lands in a location (line 4), it uses the adjusting sensing plan function (Algorithm 4) to update its ESP to its real value based on the actual participants in that cell (line 5). If its actual sensing capacity b_i has enough room to accommodate the sensing plan, then it proceeds to collect data, updates its capacity (line 6), and adds that location to its trajectory (lines 9-10).

C. ADJUSTING SENSING PLAN

Algorithm 4 updates v_i's ESP based on the actual number of participants arriving at a cell. First, the algorithm computes the sensing plans based on the total number of participants (line 3) and checks how many of these participants have enough capacity to collect (lines 5-12). If the number of eligible participants to collect is not equal to the number of participants used to compute the sensing plans (line 13), then the sensing plans are recomputed with the updated set of eligible participants. The while loop continues until the number of participants used to compute the sensing plans matches the number of participants that are eligible to collect.

V. PERFORMANCE EVALUATION

We evaluate the performance of our proposed ATNE algorithm by comparing it with the following three route generation approaches: Baseline (default AVs trajectories), greedy movement to cells with the highest reward, and weighted random walk. The reason behind selecting these three alternative approaches is that, for a fair comparison, the candidates should build trajectories based on the concept of NE sensing plan. After a careful review of the literature,

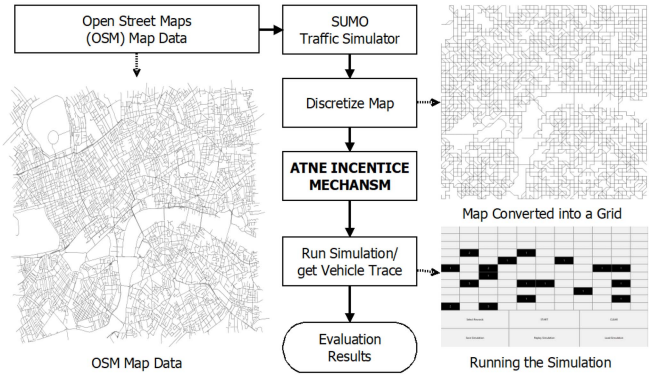


FIGURE 3. Simulation components.

we found we are pioneers in this domain. Hence, it seems natural to build the mentioned three trajectory generators. For all route's generation approaches, every participant has been assigned a sensing capacity, and uses the same data collection method. Specifically, data collection for all of the approaches is based on the computation of participant's sensing plan by Algorithm 1, and Algorithm 4.

Several experiments were designed to evaluate the approaches based on the following metrics: Crowdsourcers coverage (number of crowdsourcers from which data was collected at least once), road utilization (number of visited cells crossing roads), crowdsourcers visited (number of crowdsourcers visited at least once, regardless of whether data was collected from them or not), true positive collection rate (number of times ATNE selects a destination in the hope of collecting data, and actually collects), and average participants' utility.

A final experiment explores the effect of modeling ATNE as a complete versus an incomplete information game on metrics such as crowdsourcer coverage, road utilization, average participant's utility, and true positive collection rate. The python code implementation for all experiments can be found in https://github.com/flpolyproject/ATNE.

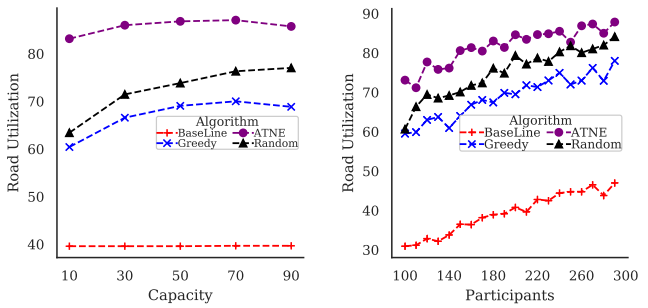
A. SIMULATION SETUP

Figure 3 depicts the elements of our simulation environment. It includes a discretized version of an Open Street Map (OSM) of London, our ATNE algorithm, an urban traffic simulator, and a visualization module.

We selected a dense map's section with drivable roads to allow variation in routing possibilities. Then, the OSM and the participant traces are linked and imported into the traffic simulator SUMO. SUMO provides realistic vehicle movements and routing algorithms on the imported map. Our map discretization process consists of transforming the OSM into a grid-based map with 8 degrees of freedom. We do that by partitioning the 5200 by 5200 squared meters section of the SUMO map into a grid with 52 by 52 cells (each cell has a size of 100 by 100 meters). Then, a vertex is placed in the center of each cell, and a directed edge is placed between the corresponding vertices based on the road direction in the actual map that crosses the geographical boundary of one cell and another. A detailed description of this procedure can be found here [30]. Figure 3 shows the difference between the original OSM map and its grid-based version. The use of this grid-based map allows us to implement our synchronous mobility model where participants moves one step at a time.

TABLE 1. Simulation parameteres.

Parameters		Parameters (cont.)	
Target Area	5200 x 5200 mt	Area Coordinates	
Cell Size	100 x 100 mt	*Low Lat*	51.480861
Deployed Sensing Task	40%	*Low Lon*	0.176478
Reward Distribution		*High Lat*	51.539079
μ	50	*High Lon*	−0.057695
σ	10	Autonomous Vehicles	
Location sensing task deployment		*Amount*	200
μ	$(0,0),(52,0),$ $(0,52),(52,52)$	*Source*	random
		Destination	(0, 0)
σ	10		
Capacity	*Experiment 1*	···	*Experiment 50*
ATNE	range(start=10,end=90,step=20)	···	—
Greedy	range(start=10,end=90,step=20)	···	—
Random	range(start=10,end=90,step=20)	···	—
Base	range(start=10,end=90,step=20)	···	—



(a) Capacity adjustment (b) Participant adjustment

FIGURE 4. Road utilization lift.

B. EXPERIMENTAL SETUP

Table 1 summarizes the simulation parameters for the experiments. We consider every cell crossed by a road as a potential location for the sensing tasks.

1) EXPERIMENT 1 (ROAD UTILIZATION)

The goal of this experiment is to explore the effects of participant's capacity, and number of participants on road network utilization. Here, we increase the value of participant capacity from 10 to 90 in multiple steps with a step-size of 20, while keeping the other parameters fixed, such as the reward value distribution, fraction of valid cells with rewards, rewards' location distribution, and number of AVs. Detailed information about the experiment parameters as well as the road network setup can be found in Table 1.

As Figure 4a shows, ATNE significantly outperforms baseline, random, and greedy approaches for different values of participant capacities. Figure 7 shows the box plots of road network utilization values for 50 trials per capacity value.

The second part of this experiment explores the relationship between number of AVs versus road utilization. In other

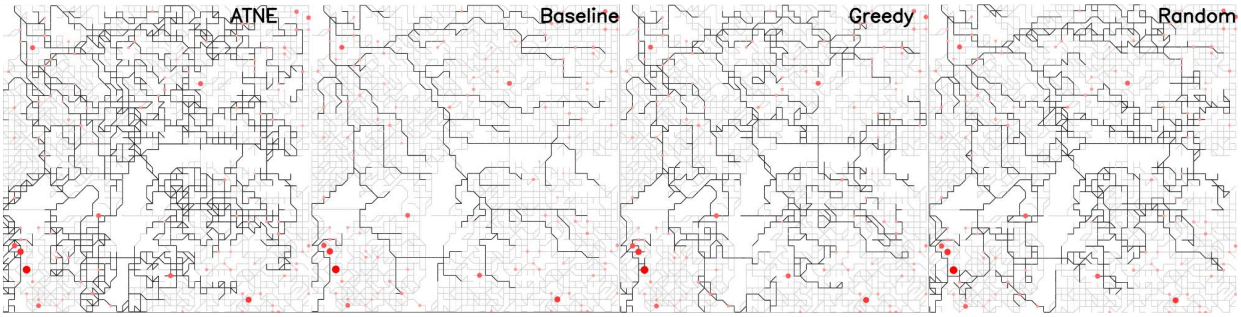
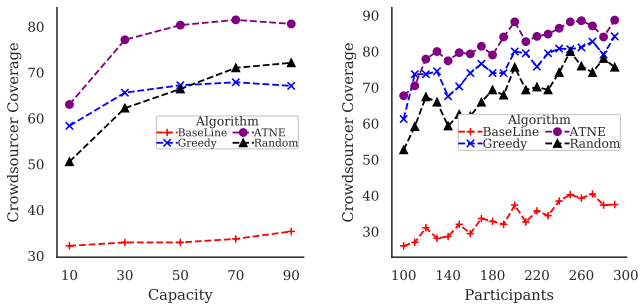


FIGURE 5. Visualization of vehicle traces and sensing tasks. Size of red dots represent reward amounts proportionally.



(a) Capacity adjustment (b) Participant adjustment

FIGURE 6. Crowdsourcers coverage lift.

words, the experiment shows how the performance of our proposed ATNE compares to other approaches for different number of vehicles. Figure 4b shows that ATNE outperforms again all the other approaches. However, as number of vehicles goes beyond 290, greedy approach performs just as good as ATNE. This phenomena is explained by the fact that after 290 vehicles the street network starts to be saturated by the high number of vehicles.

The results of the t-test in Table 2 confirm the statistical significance of the improvements of ATNE for all three metrics when compared to other methods for different number of participants.

Figure 5 visualizes the set of trajectories that result from using four different trajectory generation approaches. Here, different trajectories are overlaid on our discretized version of the map of London, with a fixed capacity of 170. The remaining parameters follow the description of Table 1. As it can be seen, participants utilized the map significantly more by using ATNE.

2) EXPERIMENT 2 (CROWDSOURCERS COVERAGE)

The goal of this experiment is to study the algorithm's ability of maximize crowdsourcer coverage. There are two main factors in pursuing this goal. First, the ability of visiting a crowdsourcer, and second, the ability of collecting from it. Big rewards attract participants, however, visiting them is not always in the best interest of a participant. An accurate estimation of the potential utilities, and sensing plans

TABLE 2. P-values comparing ATNE with greedy and random methods for crowdsource coverage and road utilization metrics.

	Crowdsourcer Coverage		Road Utilization		Average Utility	
Participants	Greedy	Random	Greedy	Random	Greedy	Random
100	$< .001$	$< .001$	0.001	$< .001$	$< .001$	$< .001$
150	$< .001$	$< .001$	0.005	$< .001$	$< .001$	$< .001$
200	$< .001$	$< .001$	$< .001$	$< .001$	$< .001$	$< .001$
250	$< .001$	$< .001$	$< .001$	$< .001$	$< .001$	$< .001$
300	$< .001$	$< .001$	$< .001$	$< .001$	$< .001$	$< .001$

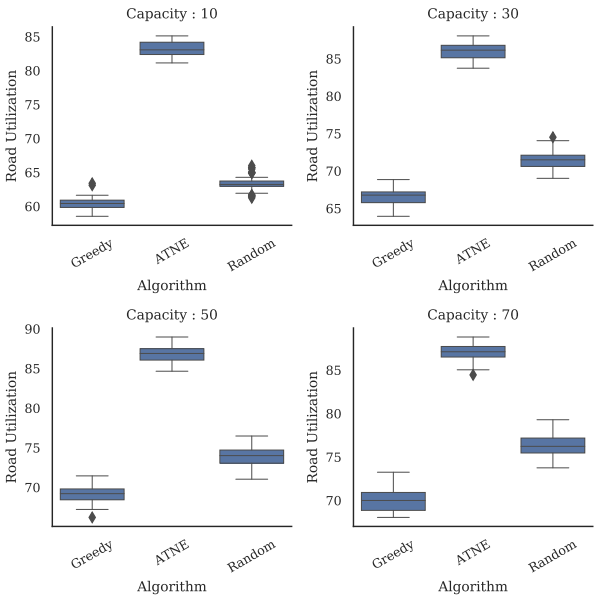
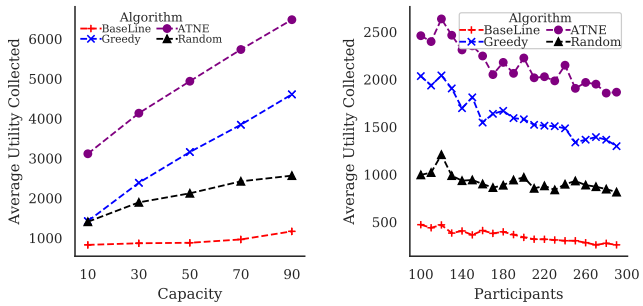


FIGURE 7. Road utilization box plots for different capacities.

(collected amount of data) allow participants to better match sensing tasks with its own constrains in terms of sensing capacity. This may result in a more profitable trajectory for a participant.

The first part of the experiment studies the relationship between participants' capacity and crowdsourcer coverage. We fix the number of AVs and the distribution of the rewards while the value of capacity is increased from 10 to 90 with a step size of 20. Figure 6a shows that crowdsourcer coverage exhibits an increasing behavior until the step size 70 for all approaches. However, after 70 a slightly decrease of curb seems to occur. Also, the same graph shows our proposed algorithm outperforms baseline, random and greedy approaches for any value of participant capacity.



(a) Capacity adjustment (b) Participant adjustment

FIGURE 8. Utility lift.

The box plots in Figure 7 show the distribution of road utilization for 50 trials per capacity value.

The second part of this experiment explores the relationship between number of AVs versus crowdsourcer coverage. Figure 6b shows that ATNE outperforms baseline, greedy, and random almost for any number of participants.

3) EXPERIMENT 3 (AVERAGE PARTICIPANT UTILITY)

The goal of this experiment is to explore the relationship between participants using ATNE, and trajectory profitability. We compute average participant utility by averaging all participants' utilities that results from collecting data during their journey from source to destination.

The first part of the experiment explores how participants' capacities affect average participants' utilities, and how our incentive mechanism compares with other approaches. In this regard, Figure 8a shows a significant lift of the average participants' utility using our proposed game theoretical approach. In particular, we see how ATNE scales linearly, namely increases in data capacity result in increases in participants' utility.

The box plots in Figure 9 show the distribution of crowdsourcer coverage for 50 trials per capacity value. We can see that for increasing values of capacity ATNE outperforms its competitors while keeping a low variability.

The second part of this experiment explores the relationship between number of participants, and average participant utility. Figure 8b shows how ATNE clearly outperforms the other trajectory generation approaches. It also shows how the participants' utility slowly decreases as a result of increments in the number of participants. This result is consistent with the collection approach used for all four approaches, namely the sensing game. Thus, as the number of participants goes up, the sensing rewards have to be shared among more participants.

4) EXPERIMENT 4 (PREDICTION)

The goal of this experiment is to explore both the ATNE's ability of predicting participant's sensing plan, and the ability of visiting or reaching a crowdsourcer. The combination of these two characteristics result in both high crowdsourcer coverage, and also a set of trajectories with high profitability.

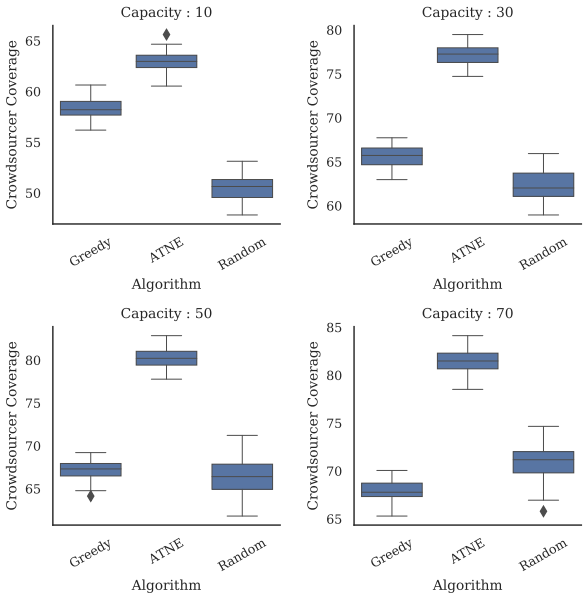
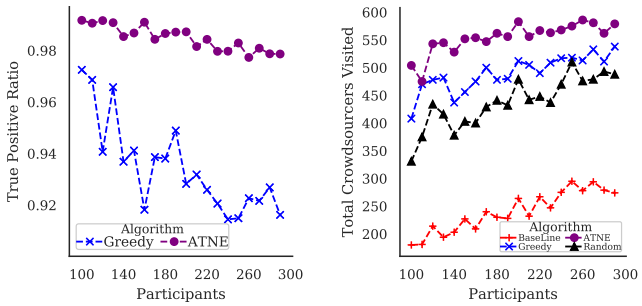


FIGURE 9. Crowdsourcer coverage box plots for different capacities.



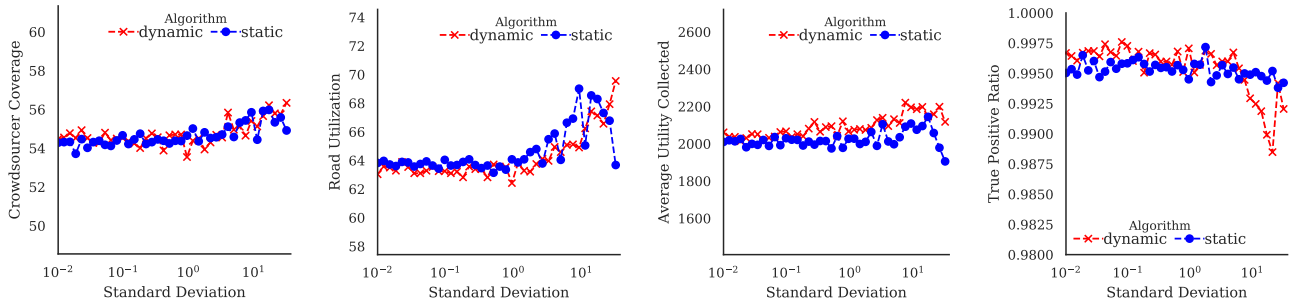
(a) True positives (TP) (b) Visited crowdsourcers

FIGURE 10. Utility encounter.

In order to make a fair comparison, we use the same collection approach for all four approaches used in this article. This approach uses Algorithm 1, and Algorithm 4 to compute the actual sensing plan (actual collection). Thus, what really distinguishes the algorithms from each other is the used mobility model (i.e., random, greedy, base line, ATNE).

While ATNE uses both expected utility and expected sensing plan to select the next destination to visit, the greedy algorithm always selects the destination offering the highest reward. In the case of ATNE, it selects the destination that yields the highest expected utility provided that participant's capacity is greater than or equal to its expected sensing plan. Thus, an accurate prediction of the actual sensing plan by the expecting sensing plan guarantees that once the participant arrives to the selected destination it will have enough capacity to collect data.

The first experiment compares the true positive data collection rates of ATNE and the greedy approach for different number of participants. True positive collection



(a) sd adjustment (b) sd adjustment (c) sd adjustment (d) sd adjustment

FIGURE 11. Complete vs Incomplete Game.

rate is the number of visited crowdsourcers from which the participant was able to collect data, divided by the total number of crowdsourcers it visited. Figure 10a shows ATNE outperforming greedy algorithm for any number of participants.

Figure 10b compares the number of visited crowdsourcers by base line, random, greedy and ATNE algorithms. Here, ATNE again outperforms all other trajectory generation approaches for almost any number of participants.

5) EXPERIMENT 5 (COMPLETE VS INCOMPLETE INFORMATION GAME)

So far, we have modeled the incentive mechanism in each local sensing game as a complete information game, where the participants are able to predict precisely other participants sensing plans as everybody sensing cost is a common knowledge. However, in real world, this assumption may not be hold, where participants have incomplete information. For instance, participants may have privacy concerns or the cost sharing information is high. Thus, the goal of this experiment is to explore the effect of incomplete information on the following metrics: crowdsensing coverage, road utilization, average participant utility, and coverage true positive rate.

Although the exact sensing costs may not be shared among participants, but given the spread and massive uses of similar mobile devices with standardized software and hardware it is reasonable to assume participants are incurred in close collection costs. As a result, the ith participant computes the ESP, and the EU of visiting crowdsourcer j by assuming all participants in j's neighborhood share the same cost, i.e., $k_i^j = k^j$ for all i.

Using this assumption and Algorithm 1, the Nash equilibrium sensing plan can be stated as

$$t_i^j = \frac{(w_j - 1) \times R_j}{w_j^2 \times k} \tag{5}$$

By replacing Equation (1) in equation (5), the utility of participant i working for crowdsourcer j becomes $u_i^j = \frac{R_j}{w_j^2}$. We use this simplified way to calculate the Nash Equilibrium sensing plan and participants' utility to compute the values for the evaluated metrics.

We first compare the behavior of crowdsourcer coverage under both complete (red) and incomplete (blue) approaches for increasing values of standard deviation of participants' sensing cost (k). Figure 11a shows that there is no significant difference in terms of this metric. Figure 11b compares the performance of both approaches on road utilization. Here, both approaches perform very similar for standard deviation values less than five. For average participant utility, Figure 11c shows that the complete game outperforms the incomplete game for a minimum difference. Finally, Figure 11d shows that for collection true positive ratio the complete game slightly outperforms the incomplete game for almost every value of standard deviation. As a result, the assumption of having the same costs for every participant working for a crowdsourcer doesn't harm the performance of the system. However, a big advantage of using this assumption is that it allows us to significantly speed up the pipeline. In fact, computing the exact ESP and EU follows binomial distribution computational burdensome for large N which is $O(2^N)$. But by assuming the same participants costs, we are equipped to use a Poisson-Binomial approximation where it is significantly less complex with $O(Nlog^2(N))$.

VI. CONCLUSION

In this article, we introduced the design and evaluation of an incentive mechanism for vehicular crowdsensing (VCS). We modeled the incentive mechanism as a non-cooperative game. We formulated the participants' trajectory Nash Equilibrium problem and proposed a greedy algorithm to approximate the solution. Through the use of real-world traffic maps and extensive simulations (SUMO), we evaluated the performance of the proposed mechanism in terms of spatial coverage, road utilization, and participants utility. We also discussed the limitations and challenges of the proposed method throughout this article. Specifically, we address the impact of using realistic assumptions about the participants information sharing on the evaluation metrics, and the time complexity.

ACKNOWLEDGMENT

Any opinions, findings and conclusions, or recommendations expressed in this material are those of the author(s), and do not necessarily reflect those of the NSF.

REFERENCES

[1] F. Restuccia, S. K. Das, and J. Payton, "Incentive mechanisms for participatory sensing: Survey and research challenges," *ACM Trans. Sens. Netw.*, vol. 12, no. 2, p. 13, 2016.

[2] L. G. Jaimes, I. J. Vergara-Laurens, and A. Raij, "A survey of incentive techniques for mobile crowd sensing," *IEEE Internet Things J.*, vol. 2, no. 5, pp. 370–380, Oct. 2015.

[3] L. Wang, D. Zhang, Y. Wang, C. Chen, X. Han, and A. M'hamed, "Sparse mobile crowdsensing: Challenges and opportunities," *IEEE Commun. Mag.*, vol. 54, no. 7, pp. 161–167, Jul. 2016.

[4] N. Maisonneuve, M. Stevens, M. E. Niessen, and L. Steels, "NoiseTube: Measuring and mapping noise pollution with mobile phones," in *Information Technologies in Environmental Engineering*. Heidelberg, Germany: Springer, 2009, pp. 215–228.

[5] Z. He, J. Cao, and X. Liu, "High quality participant recruitment in vehicle-based crowdsourcing using predictable mobility," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Kowloon, Hong Kong, 2015, pp. 2542–2550.

[6] S. Xu, X. Chen, X. Pi, C. Joe-Wong, P. Zhang, and H. Y. Noh, "Incentivizing vehicular crowdsensing system for large scale smart city applications," in *Proc. Sens. Smart Struct. Technol. Civil Mech. Aerosp. Syst.*, vol. 10970, 2019, Art. no. 109701C.

[7] A. S. El-Wakeel, J. Li, A. Noureldin, H. S. Hassanein, and N. Zorba, "Towards a practical crowdsensing system for road surface conditions monitoring," *IEEE Internet Things J.*, vol. 5, no. 6, pp. 4672–4685, Dec. 2018.

[8] C. Wang, Z. Zhang, L. Shao, and M. Zhou, "Estimating travel speed via sparse vehicular crowdsensing data," in *Proc. IEEE 3rd World Forum Internet Things (WF-IoT)*, Reston, VA, USA, 2016, pp. 643–648.

[9] S. J. Sheikh, A. Basalamah, H. Aly, and M. Youssef, "Demonstrating map++: A crowd-sensing system for automatic map semantics identification," in *Proc. IEEE Annu. Int. Conf. Sens. Commun. Netw. (SECON)*, Singapore, 2014, pp. 152–154.

[10] X. Zhu, S. A. Samadh, and T. Yu, "Large scale active vehicular crowdsensing," in *Proc. IEEE Veh. Technol. Conf. (VTC)*, Chicago, IL, USA, 2018, pp. 1–5.

[11] K. Shilton *et al.*, "Participatory design of sensing networks: Strengths and challenges," in *Proc. 10th Anniversary Conf. Participatory Design*, 2008, pp. 282–285.

[12] Y. Tong, L. Chen, and C. Shahabi, "Spatial crowdsourcing: Challenges, techniques, and applications," *Proc. VLDB Endowment*, vol. 10, no. 12, pp. 1988–1991, 2017.

[13] D. Braess, A. Nagurney, and T. Wakolbinger, "On a paradox of traffic planning," *Transp. Sci.*, vol. 39, no. 4, pp. 446–450, 2005.

[14] Y. Li and A. S. Morse, "The power allocation game on a network: A paradox," *IEEE/CAA J. Autom. Sinica*, vol. 5, no. 4, pp. 771–776, Jul. 2018.

[15] E. Koutsoupias and C. Papadimitriou, "Worst-case equilibria," in *Proc. Annu. Symp. Theor. Aspects Comput. Sci.*, 1999, pp. 404–413.

[16] L. Xue, C. Sun, D. Wunsch, Y. Zhou, and F. Yu, "An adaptive strategy via reinforcement learning for the prisoner's dilemma game," *IEEE/CAA J. Autom. Sinica*, vol. 5, no. 1, pp. 301–310, Jan. 2017.

[17] L. G. Jaimes, A. Chakeri, J. Lopez, and A. Raij, "A cooperative incentive mechanism for recurrent crowd sensing," in *Proc. SoutheastCon*, Fort Lauderdale, FL, USA, 2015, pp. 1–5.

[18] J. Lu, Y. Xin, Z. Zhang, X. Liu, and K. Li, "Game-theoretic design of optimal two-sided rating protocols for service exchange dilemma in crowdsourcing," *IEEE Trans. Inf. Forensics Security*, vol. 13, pp. 2801–2815, 2018.

[19] D. Yang, G. Xue, X. Fang, and J. Tang, "Incentive mechanisms for crowdsensing: Crowdsourcing with smartphones," *IEEE/ACM Trans. Netw.*, vol. 24, no. 3, pp. 1732–1744, Jun. 2016.

[20] A. Chakeri and L. G. Jaimes, "An incentive mechanism for crowdsensing markets with multiple crowdsourcers," *IEEE Internet Things J.*, vol. 5, no. 2, pp. 708–715, Apr. 2018.

[21] A. Chakeri and L. G. Jaimes, "An iterative incentive mechanism design for crowd sensing using best response dynamics," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Paris, France, 2017, pp. 1–7.

[22] V. S. Dasari, B. Kantarci, M. Pouryazdan, L. Foschini, and M. Girolami, "Game theory in mobile CrowdSensing: A comprehensive survey," *Sensors*, vol. 20, no. 7, p. 2055, 2020.

[23] X. Liang and Z. Yan, "A survey on game theoretical methods in human–machine networks," *Future Gener. Comput. Syst.*, vol. 92, pp. 674–693, Mar. 2019.

[24] W. Xu *et al.*, "Internet of Vehicles in big data era," *IEEE/CAA J. Autom. Sinica*, vol. 5, no. 1, pp. 19–35, Jan. 2018.

[25] N. Lu, N. Cheng, N. Zhang, X. Shen, and J. W. Mark, "Connected vehicles: Solutions and challenges," *IEEE Internet Things J.*, vol. 1, no. 4, pp. 289–299, Aug. 2014.

[26] K. Han, C. Chen, Q. Zhao, and X. Guan, "Trajectory-based node selection scheme in vehicular crowdsensing," in *Proc. IEEE/CIC Int. Conf. Commun. China (ICCC)*, Shenzhen, China, 2015, pp. 1–6.

[27] Y. Chen, P. Lv, D. Guo, T. Zhou, and M. Xu, "Trajectory segment selection with limited budget in mobile crowd sensing," *Pervasive Mobile Comput.*, vol. 40, pp. 123–138, Sep. 2017.

[28] K. Kong, J. Xu, M. Xu, L. Tu, Y. Wu, and Z. Chen, "Trajectory query based on trajectory segments with activities," in *Proc. 3rd ACM SIGSPATIAL Workshop Smart Cities Urban Anal.*, 2017, p. 2.

[29] D. Yang, G. Xue, X. Fang, and J. Tang, "Crowdsourcing to smartphones: Incentive mechanism design for mobile phone sensing," in *Proc. 18th Annu. Int. Conf. Mobile Comput. Netw.*, 2012, pp. 173–184.

[30] Q. Goss, M. İ. Akbaş, L. G. Jaimes, and R. Sanchez-Arias, "Street network generation with adjustable complexity using k-means clustering," in *Proc. SoutheastCon*, Huntsville, AL, USA, 2019, pp. 1–6.